

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the Application of:

Zhao et al.

Application Serial No.: 10/017,495

Filing Date: December 14, 2001

For: NAME SERVICE OBJECT CLUSTERING

Examiner: Lechi Truong

Group Art Unit: 2194

Date: June 1, 2009

Atty. Docket No.: BOR-160

CERTIFICATE OF TRANSMISSION

I hereby certify that this correspondence is transmitted via EFS-Web to the United States Patent and Trademark Office on June 1, 2009.

Signed: 
Charisse E. Leong

APPELLANTS' PAPER COMPLYING WITH 37 C.F.R. § 41.37(c)(1)(v)

Mail Stop Brief – Patents
Commissioner for Patents
P.O. Box. 1450
Alexandria, VA 22313-1450

Dear Sir:

Appellants submit this paper in accordance with 37 C.F.R. § 41.37(c)(1)(v) in support of their appeal from the final Office Action mailed September 7, 2005 by Examiner Lechi Truong in the above-identified patent application.

MPEP 1205.03 states, “When the Office holds the brief to be defective solely due to appellant's failure to provide a summary of the claimed subject matter as required by 37 CFR 41.37(c)(1)(v), an entire new brief need not, and should not, be filed. Rather, a paper providing a summary of the claimed subject matter as required by 37 CFR 41.37(c)(1)(v) will suffice.”

SUMMARY OF CLAIMED SUBJECT MATTER

Apellants' invention as claimed is summarized and explained below with reference numerals, specification page numbers, and drawing figure numbers indicating where the claim involved with the present appeal finds support in the specification and the drawings.

1. A computer implemented method for fault tolerance, load balance and failover of CORBA object servers, comprising the steps of:

establishing name service clusters for the object servers which each contain a unique object binding table that contains object server references; **(FIG. 3) [page 7; lines 2-4]**

in response to a request from a client that invokes a cluster, performing a load balance by having the name service select an object server located in the invoked cluster; **(500; 510, 520; FIG. 5) [page 9-10; lines 21-1]**

appending a cluster component to the invoked cluster to provide failover upon failure of the object server; **(530; FIG. 5) [page 10; lines 1-3]**

forwarding a selected object reference to a client upon completion of the load balance; and **[page 4; lines 2-3]**

permitting the client to communicate with the server associated with the selected object reference which was forwarded to the client, wherein the fault tolerance, the load balance and the failover are performed transparently. **(6; FIG. 3) [pages 8-9; lines 21-3]**

2. The method of claim 1, said invoking step comprising the step of:

binding to the server using an IP Address and port number contained in the specific object reference. **(1; FIG. 4) [page 7; lines 22-24]**

9. The method of claim 1, further comprising the step of:

specifying a load balance algorithm upon establishing the naming service cluster to perform name service load balancing of object references contained within the clusters. **(530; FIG. 5) [page 10; lines 1-3]**

10. The method of claim 1, wherein said load balancing is performed based on a predetermined method. **[page 5; lines 15-17]**

12. The method of claim 1, wherein said load balancing is performed based on a predetermined method. **[page 5; lines 15-17]**

13. The method of claim 12, wherein the predetermined method is a Round robin load balancing algorithm. **[page 5; lines 15-17]**

14. The method of claim 1, wherein the object binding table of each cluster contains object references;

wherein each object server reference represents a single server. **[page 7; lines 2-4]**

15. A computer implemented method for fault tolerance, load balance and failover of CORBA object servers, comprising the steps of:

establishing name service clusters for the object servers which each contain a unique object binding table that contains object server references; **(FIG. 3) [page 7; lines 2-4]**

setting a flag in a file to activate the implicit clustering; **[page 5; lines 10-11]**

in response to invoking a cluster contained in a context having clusters;

performing a load balance by having the name service select an object server located in the clusters; **(600; 610, 620, 630; FIG. 6) [page 10; lines 7-14]**

forwarding a selected object reference to a client upon completion of the load balance; and **[page 4; lines 8-9]**

communicating with the server associated with the selected object reference which was forwarded to the client, wherein the fault tolerance, the load balance and the failover are performed transparently. **(6; FIG. 3) [pages 8-9; lines 21-3]**

16. The method of claim 15, wherein the file is a configuration file. **[page 5; lines 10-11]**

17. The method of claim 15, said invoking step comprising:

binding to the server using an IP Address and port number contained in the specific object reference. **(1; FIG. 4) [page 7; lines 22-24]**

18. The method of claim 15, wherein said load balancing is performed based on a

predetermined method. **[page 5; lines 15-17]**

19. The method of claim 18, wherein the predetermined method is a Smart Round Robin load balancing algorithm. **[page 5; lines 15-17]**

20. The method of claim 15, wherein object reference binding having identical names are clustered together in common clusters such that a common group of object reference binders servers is created. **[page 9; lines 4-6]**

21. The method of claim 20, further comprising the step of:
specifying a load balance algorithm to perform load balancing of object references contained within the common group of group of object reference binders. **(620; FIG. 6) [page 10; lines 10-11]**

22. The method of claim 21, wherein initially the load balance algorithm is Smart Round Robin. **[page 5; lines 15-17]**

23. A computer implemented method for transparently load balancing CORBA object servers, comprising the steps of:

establishing name service clusters for the object servers which each contain a unique object binding table that contains object server references; **(FIG. 3) [page 7; lines 2-4]**

in response to a request from a client that invokes a cluster, the name service for the cluster performs a load balance to select an object server located in the invoked cluster which can handle the request; **(500; 510, 520; FIG. 5) [page 9-10; lines 21-1]**

forwarding a selected object reference to a client upon completion of the load balance; and **[page 4; lines 2-3]**

permitting the client to communicating with the server associated with the selected object reference that was forwarded to the client. **(6; FIG. 3) [pages 8; lines 25-6]**

24. The method of claim 23 further including the step of appending a cluster component to the invoked cluster, such cluster providing information to the client as to a fail over server to be accessed upon failure of the object server. **(530; FIG. 5) [page 10; lines 1-3]**

Respectfully submitted,

A handwritten signature in black ink, appearing to read 'S. Kokka', with a stylized flourish at the end.

Scott S. Kokka
Reg. No. 51,893

KOKKA & BACKUS, PC
200 Page Mill Road, Suite 103
Palo Alto, CA 94306-2022
Tel: (650) 566-9921
Fax: (650) 566-9922